High Precision Calibration Method for Asynchronous time-to-Digital Converters

Wassim Khaddour, Wilfried Uhring, Foudil Dadouche, Norbert Dumas, Morgan Madec

High precision calibration method for asynchronous time-to-digital converters

Wassim Khaddour, Wilfried Uhring, Foudil Dadouche, Norbert Dumas, Morgan Madec
ICube Laboratory
University of Strasbourg and CNRS, Strasbourg, France
wilfried.uhring@unistra.fr

***Abstract*— Field-programmable gate array (FPGA)-based time-to-digital converters (TDCs) suffer from large bin width variations. This issue imposes performing a calibration process to compensate the nonlinearity of the TDC. The most commonly used calibration technique is the bin–by-bin calibration that can improve the differential nonlinearity (DNL) of the TDC up to 10 times. In this paper, we propose a new robust calibration method for asynchronous TDCs and compare it with the bin-by-bin method. The simulation results showed that the proposed method is less sensitive to the nonlinearity of the TDC, and an improvement of more than 100 times can be achieved in regard to the DNL of the non-calibrated TDC. These results have been confirmed by experimental measurements made on an asynchronous TDC implemented on a Cyclone V SoC-FPGA kit. Density code tests were performed to measure the root mean square (RMS) DNL of the TDC and those of the calibrated histograms for the two methods. For a TDC with a DNL of 0.71 least significant bit (LSB), the DNL of the calibrated histogram is 0.053 LSB for the classical bin-by-bin method and 0.005 LSB, i.e., 10 times better, for the proposed method.**

Keywords—field-programmable gate array (FPGA), time-to-digital converter (TDC), calibration techniques, differential nonlinearity (DNL), integral nonlinearity (INL).

I. Introduction

Time-to-digital converters (TDCs) are used for a wide range of applications to measure precise time intervals [1-4]. In general, application-specific integrated circuits (ASICs) are high-performance solutions to build high-resolution TDCs. However, field-programmable gate array (FPGA) can be a better choice for many applications because of their flexibility and short development time [5-7]. Furthermore, system-on-chip FPGA (SoC-FPGA) kits integrate a hard processor system with FPGA fabric which enables the efficient integration of a downstream processing such as a post calibration process [6].

The Coarse-Fine architecture is widely adopted in FPGA-based TDCs to increase the full-scale range (FSR) and to improve the resolution [8]. Usually, the coarse TDC is a simple counter running at the system clock frequency, whereas the fine TDC is commonly based on the time interpolation technique [9]. The most used architecture to build the fine TDC is based on the tapped delay line (TDL) technique. However, this architecture suffers from large variations in the delay of the TDL cells which induce time nonlinearity of the TDC [10]. To overcome this problem, many research works have presented calibration methods to improve the linearity of the FPGA-based TDCs. Most of these works addressed synchronous TDCs, in which the start signal is synchronous to the system clock and have a coarse and only one fine TDC, but they didn't cover the asynchronous ones, that include two fine TDCs. This paper presents a robust method for the calibration of asynchronous TDCs. This method is a post-processing on the raw data of FPGA-based or ASIC TDCs. To evaluate this method, it will be compared with the bin-by-bin calibration method which is the most used method [1][3][4][8] [11].

This paper is organized as follows: Section II is a background about the general functional principle of Coarse-Fine asynchronous TDCs and the bin-by-bin calibration method. Section III presents the proposed calibration method "Matrix calibration", starting by introducing the average-bin-width calibration method that is used for synchronous TDCs. The simulation and experimental results are presented in sections IV and V respectively. Finally, section VI is a conclusion of this paper.

II. Background: Asynchronous TDCs and Bin-by-bin Calibration

A. Asynchronous TDC

For the Coarse-Fine asynchronous TDCs, the start and the stop signals are both asynchronous to the TDC clock. The time interval between the start and stop signals can be calculated by the following equation as illustrated in Figure 1:

$$T_m = T_{coarse} + T_{fine2} - T_{fine1} \quad (1)$$

Where T_{fine1} is the interval between the start signal and the system previous clock rising edge, T_{fine2} is the interval between the stop signal and the previous clock rising edge and T_{coarse} is the number of system clock cycles between the two signals multiplied by the clock period.

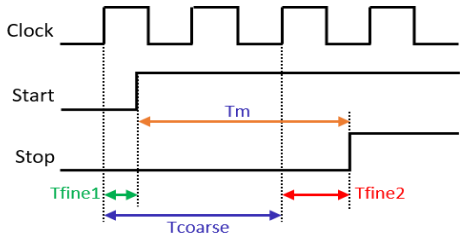

Figure 1 Time interval measurement using an asynchronous TDC. Two fine interval measurements are required for the Start and the Stop signals in addition to the coarse measurement.

Thus, A Coarse-Fine asynchronous TDC should contain two fine TDCs, the start fine TDC for the measurement of T_{fine1} and the stop fine TDC for the measurement of T_{fine2}, and a coarse counter that measures T_{coarse}.

B. Bin-by-bin calibration method

In this method, a code density test, detailed in the next section, is performed to determine the widths of the TDC bins and then to calculate the calibrated time of each bin, which corresponds to the center of this bin, as following:

$$\tau(i) = \left[\frac{N(i)}{2} + \sum_{k=0}^{i-1} N(k)\right] * \frac{T}{N} \quad (2)$$

Where $\tau(i)$ is the calibrated time of the bin i, $N(i)$ is the number of counts in the bin i, $N(k)$ is the number of counts in the bin k and N is the total number of counts and T is the total delay of the TDC.

The calibrated time of the bins are stored in a lookup table (LUT) that will be used to convert the raw bin number, i.e., the output of the TDC's encoder, to the calibrated bin number.

III. Matrix Calibraion

A. Average-bin-width calibration

This method can be used to calibrate synchronous TDCs. A code density test is executed to determine the delays of the bins along the delay line. The number of counts in each bin of the code density histogram is related to the width of this bin. If the total number of counts is N, the delay of the bin b will be:

$$T(b) = \frac{N(b)}{N} * T \qquad (3)$$

Where $N(b)$ is the number of counts of the bin b and T is the total delay of the TDC, i.e., its FSR.

The aim of this calibration is to have M calibrated bins with a uniform time width Tc that can be calculated as follows:

$$Tc = \frac{T}{M} \qquad (4)$$

Hence, the number of counts in the calibrated bin Nc will be:

$$Nc = \frac{N}{M} \qquad (5)$$

To simplify the explanation of this calibration technique, we consider a simple code density histogram, shown in Figure 2(a), with 5 non-identical bins called the raw bins. To create a calibrated histogram with 5 identical calibrated bins, the total counts of the code density histogram should be uniformly redistributed on the 5 calibrated bins. The percentage share of each raw bin in the calibrated ones is calculated and saved in a table called the calibration table, as illustrated in Figure 2(b). Thereafter, when performing a measurement, this table will be used to create the calibrated histogram from the measurement raw histogram.

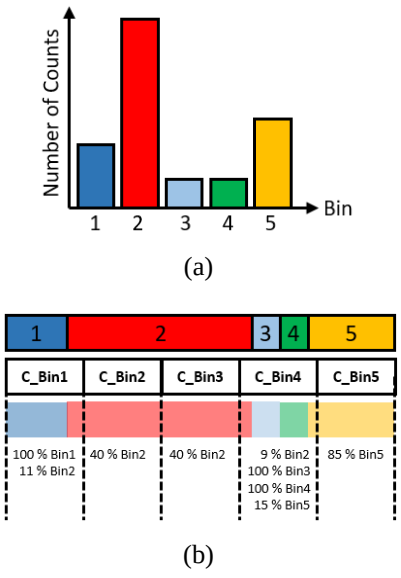

Figure 2 Average-bin-width calibration, (a) Code density raw histogram, (b) Calibration table, that describes the share percentages of the raw bins in the calibrated bins.

Furthermore, the number of calibrated bins is configurable and the time resolution of the TDC can be adjusted according to the selected number of calibrated bins. For L calibrated bins the time resolution will be:

$$Tc = \frac{T}{L} \qquad (6)$$

And the number of counts in the calibrated bin:

$$Nc = \frac{N}{L} \qquad (7)$$

B. Matrix calibration

This method is based on the Average-bin-width calibration method and used to calibrate Coarse-Fine asynchronous TDCs. As explained in II-A, each event is described by three factors: T_{fine1}, $T_{fine2\ and}$ T_{coarse}. Thus, for this type of TDCs the raw histogram can be built as a 3-d array, cf. Figure 3.

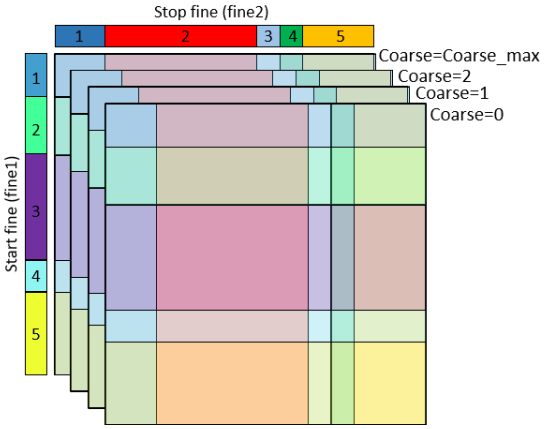

Figure 3 Three-dimension raw histogram for asynchronous TDCs, x and y dimensions represent the Stop and Start fine bins respectively, and z dimension represents the Coarse value.

If the coarse value is ignored when performing a code density test, all the counts will be accumulated in a 2-d histogram, as represented in Figure 4. Each cell of this histogram saves the number of counts for which the start and the stop signals arrive in certain start and stop fine bins. For example, the cell (2,3) holds the number of counts for which the start signal arrives in the third bin of the start fine TDC and the stop signal arrives in the second bin of the stop fine TDC. Since the fine start and fine stop TDCs have nonuniform raw bins, the size of the cell, that represents its number of counts, is different from one cell to another according to the width of its corresponding start and stop raw bins.

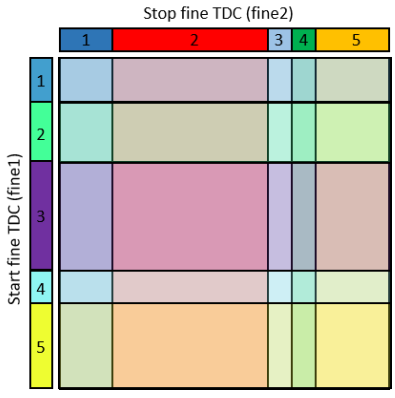

Figure 4 Two-dimension Code density histogram, the Start and Stop fine bins are accumulated for all the coarse values.

The matrix calibration method can be summarized by the even distribution of the counts of this histogram on calibrated cells in a way that all the cells have a uniform size. This can be achieved in three steps:

1) Individual calibration of Start and Stop fine TDCs:
From the 2-d code density histogram, all the counts are accumulated in two 1-d code density histograms, one for each fine TDC. These histograms are used to build the calibration tables of the start and stop fine TDCs, as explained for the Average-bin-width calibration

2) Columns calibration:
In fact, each column in the 2-d code density histogram represents a 1-d code density histogram of the start fine TDC. This histogram contains the start counts related to the events for which the stop signal arrives in the stop bin that has the same number as the considered column. Considered as a 1-d histogram, each column is separately calibrated using the calibration table of the start fine TDC. As a result of this step, all the cells will have an identical row height whereas the columns width is still nonuniform, as shown in Figure 5, and this will be corrected in the next step.

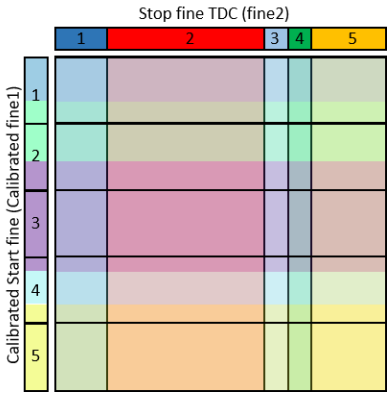

Figure 5 Columns calibration, the average-bin-width calibration is applied to the columns.

3) Rows calibration:
In this step, each row of the 2-d histogram resulted from the first step can be consider as a 1-d histogram of the stop fine TDC. Each row histogram is calibrated using the calibration table of the stop fine TDC. The output of this step is the calibrated 2-d histogram in which all the cells have the same width and height. This leads to a global uniform size of the calibrated histogram cells as shown in Figure 6.

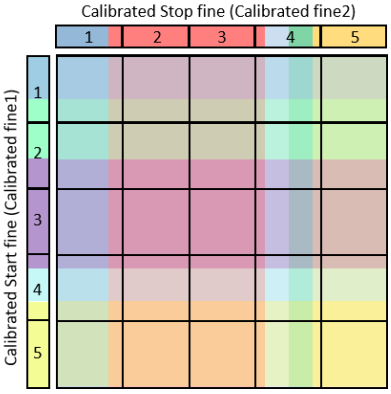

Figure 6 Calibrated two-dimension histogram, the average-bin-width calibration is applied to the rows resulting in identical cells sizes.

The previous two steps are applied to the 2-d code density histogram to prove the efficiency of the calibration method and show that all the bins in the 2-d calibrated histogram have the same size. Later on, when performing a measurement, the 3-d raw histogram, shown in figure 3 above, will be calibrated by the columns and rows Calibration for all the coarse values (0, 1, 2 ... coarse_max) to have a 3-d calibrated histogram.

4) Bulding 1-d calibrated histogram:
Each cell of the 3-d calibrated histogram is a part of a bin in the final 1-d calibrated histogram, the number of this bin is calculated by the following equation:

$$Bin\_number = C\_fine2 + (Coarse * M) - C\_fine1 \quad (8)$$

Where *C_fine1*, *C_fine2* are the row and column numbers of the calibrated cell, that represent the start and the stop corrected bin number, *Coarse* is the coarse number of the calibrated cell and *M* is the total number of calibrated bins in the stop fine TDC. For example, with $M = 5$, the number of counts of the cell ($C\_fine1 = 2$, $C\_fine2 = 4$, $coarse = 3$) should be added to the bin number (4 + (3*5) - 2 = 17) of the final calibrated 1-d histogram.

Consequently, after performing the matrix calibration, the calibrated 1-d histogram is built by scanning all the cells of the 3-d calibrated histogram and adding the counts of each cell to its corresponding bin in the 1-d calibrated histogram.

IV. Simulation Results

To test our calibration method, simulations are carried out by constructing start and stop fine TDCs in MATLAB. Each simulated TDC has 256 delay elements and a total delay of 5 ns. Furthermore, the time distribution profile of the simulated TDCs is similar to that of a real TDC implemented on a Cyclone V FPGA kit following the methodology explained in [6]. Firstly, in order to build the LUTs of the bin-by-bin calibration and the calibration tables of the matrix calibration, a code density test is performed by simulating 10^7 random events, with a start and stop arrival times uniformly distributed over the FSR of the start and stop TDCs. Next, to evaluate and compare the calibration methods, a new code density test is simulated with the same number of events. The start signal of these simulated events uniformly covers the FSR of the start fine TDC, whereas the stop signal arrives after the start signal with a random delay from 0 to 5 ns. Thus, the stop signal covers the FSR of the stop fine TDC with a maximum coarse value of 1.

1) For the bin-by-bin calibration method:
The time interval for each event is firstly calculated from the LUTs of the start and stop fine TDCs and the coarse counter. Then, a calibrated histogram is built by adding the counts to the bin that corresponds to their time interval.

2) For the matrix calibration method:
A 3-d raw histogram is built by adding the counts to their corresponding cells according to the start fine, stop fine and coarse values of each event. After that, the last three steps of the matrix calibration are applied to have a 1-d calibrated histogram.

These simulation steps are repeated for 10 pairs of simulated TDC with different RMS DNL, that varies from 0 to 1 LSB, and the RMS DNL of the calibrated histogram is measured for each method. The results, illustrated in Figure 7, show that for the bin-by-bin calibration method, the DNL of the calibrated histogram is improved by factor 10 and

increases linearly with the DNL of the TDC. Whereas the proposed method is much less sensitive to the noise of the raw TDC with an almost flat response. The first simulated TDC is an ideal one with a DNL of 0 LSB. It is obvious that, in this case, the DNL of the calibrated histogram should also be 0 LSB. Nevertheless, the DNL of the calibrated histogram obtained by the simulation results is not NULL. The reason for this is that the code density test is limited by the shot noise which can be calculated as follows:

$$Shot\ noise = \sqrt{\frac{Number\ of\ Bins}{Counts\ number}}\ LSB \qquad (9)$$

In our case, this equation gives about 0.005 LSB.

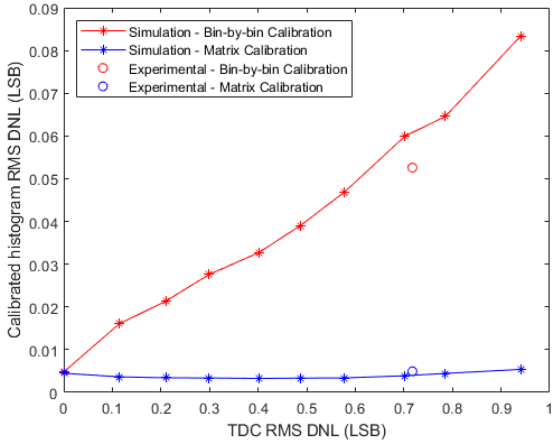

Figure 7 Simulation and experimental results of the calibrated histogram DNL versus the raw TDC DNL.

In the next simulation, start and stop fine TDCs of 256 delay elements is simulated using the time distribution of a real asynchronous TDC implemented on a Cyclone V SoC-FPGA [6]. A code density test is simulated to build the LUTs and the calibration tables in the same way as in the previous simulation. Another code density test is simulated to apply and evaluate the two calibration methods. Figure 8 illustrates the DNL and INL values of the calibrated histograms obtained for the two methods. These results show that the proposed method is up to 10 time better than the classical bin-by-bin calibration.

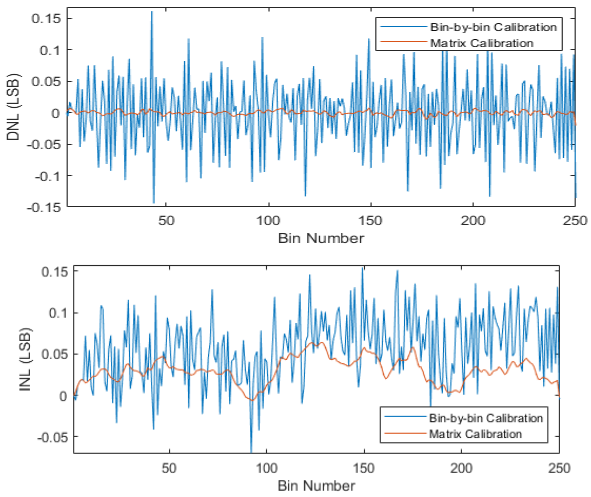

Figure 8 DNL and INL values of the calibrated histograms for the two calibration methods.

V. Experimental Results

To experimentally verify our proposed calibration method and compare it with the bin-by-bin method, we implemented an asynchronous double-TDL TDC on a cyclone V SoC-FPGA kit. Each TDL consists of 256 delay elements, and the system clock frequency is 200 MHz. The start signal is generated from asynchronous source using a separated on-chip PLL whereas the stop signal is provided by a single-photon avalanche diode (SPAD). For the calibration, a code density test is performed by connecting both the start and the stop signal to the SPAD which is exposed to ambient light. Thus, the start and stop signals arrive randomly and cover all the FSR of the TDCs. With the same number of events as in the simulation, i.e.,10^7 events, we built the LUTs for the bin-by-bin calibration and the calibration tables for the matrix calibration. We also calculated the RMS DNL of the start and stop TDCs which was 0.71 LSB. Thereafter, we performed another test with the start signal connected to the asynchronous source and the stop signal provided by the SPAD. After applying the two calibration methods, the RMS DNL of the calibrated histograms was calculated. For the bin-by-bin calibration, the RMS DNL of the calibrated histogram is 0.053 LSB, whereas it is 0.005 LSB for the matrix calibration, as indicated in Figure 7.

VI. Conclusion

In this work, a new calibration method for asynchronous TDCs is presented and compared with the most commonly used method which is the bin-by-bin calibration. Simulation results showed that the proposed method is less sensitive to the DNL of the raw TDC and up to 10 time better than the classical bin-by-bin calibration. This improvement is done at the cost of about 10 times more complex signal processing due to more multiplication and memory access instructions. The simulation results were confirmed by experiments made to calibrate a real TDC implemented on a Cyclone V SoC-FPGA.

Acknowledgment

The authors would like to thank the French National Research Agency (ANR-15-CE11-0006) and the European Interreg WPS Project for funding this study.

References

[1] K. -J. Choi and D. -W. Jee, "Design and Calibration Techniques for a Multichannel FPGA-Based Time-to-Digital Converter in an Object Positioning System," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-9, 2021, Art no. 2000209, doi: 10.1109/TIM.2020.3011490.

[2] Wang, W., Zhou, H., & Xiong, P. (2017). A TDC based on Carry-in Lines of the FPGA.

[3] J. Y. Won, S. I. Kwon, H. S. Yoon, G. B. Ko, J. Son and J. S. Lee, "Dual-Phase Tapped-Delay-Line Time-to-Digital Converter With On-the-Fly Calibration Implemented in 40 nm FPGA," in IEEE Transactions on Biomedical Circuits and Systems, vol. 10, no. 1, pp. 231-242, Feb. 2016, doi: 10.1109/TBCAS.2015.2389227.

[4] H. Chen, Y. Zhang and D. D. Li, "A Low Nonlinearity, Missing-Code Free Time-to-Digital Converter Based on 28-nm FPGAs With Embedded Bin-Width Calibrations," in IEEE Transactions on Instrumentation and Measurement, vol. 66, no. 7, pp. 1912-1921, July 2017, doi: 10.1109/TIM.2017.2663498.

[5] J. Wang, S. Liu, Q. Shen, H. Li and Q. An, "A Fully Fledged TDC Implemented in Field-Programmable Gate Arrays," in IEEE Transactions on Nuclear Science, vol. 57, no. 2, pp. 446-450, April 2010, doi: 10.1109/TNS.2009.2037958.

[6] W. Khaddour, F. Dadouche, W. Uhring, V. Frick and M. Madec, "Design Methodology and Timing Considerations for implementing a TDC on a Cyclone V FPGA Target," 2020 18th IEEE International New Circuits and Systems Conference (NEWCAS), 2020, pp. 126-129, doi: 10.1109/NEWCAS49341.2020.9159812.

[7] F. Dadouche, T. Turko, I. Malass, A. Skilitsi , J. Léonard , W. Uhring, "Design, Implementation and Characterization of Time-to-Digital Converter on Low-Cost FPGA", Sensors and Applications in Measuring and Automation Control Systems (Book Series: Advances in Sensors: Reviews), Sergey Y. Yurish (Eds.), Chapitre. 11, pages 205-229, IFSA, Volume 4, décembre 2016. http://www.sensorsportal.com/HTML/BOOKSTORE/Advance_in_Sensors_Vol_4.htm. ISBN: 978-84-617-7596-5.

[8] F. Garzetti, N. Corna, N. Lusardi and A. Geraci, "Time-to-Digital Converter IP-Core for FPGA at State of the Art," in IEEE Access, vol. 9, pp. 85515-85528, 2021, doi: 10.1109/ACCESS.2021.3088448.

[9] E. Bayer and M. Traxler, "A High-Resolution (< 10 ps RMS) 48-Channel Time-to-Digital Converter (TDC) Implemented in a Field Programmable Gate Array (FPGA)," in IEEE Transactions on Nuclear Science, vol. 58, no. 4, pp. 1547-1552, Aug. 2011, doi: 10.1109/TNS.2011.2141684.

[10] Chen, Yuan-Ho. (2018). Run-time calibration scheme for the implementation of a robust field-programmable gate array-based time-to-digital converter. International Journal of Circuit Theory and Applications. 47. 10.1002/cta.2571.

[11] W. Pan, G. Gong and J. Li, "A 20-ps Time-to-Digital Converter (TDC) Implemented in Field-Programmable Gate Array (FPGA) with Automatic Temperature Correction," in IEEE Transactions on Nuclear Science, vol. 61, no. 3, pp. 1468-1473, June 2014, doi: 10.1109/TNS.2014.2320325.